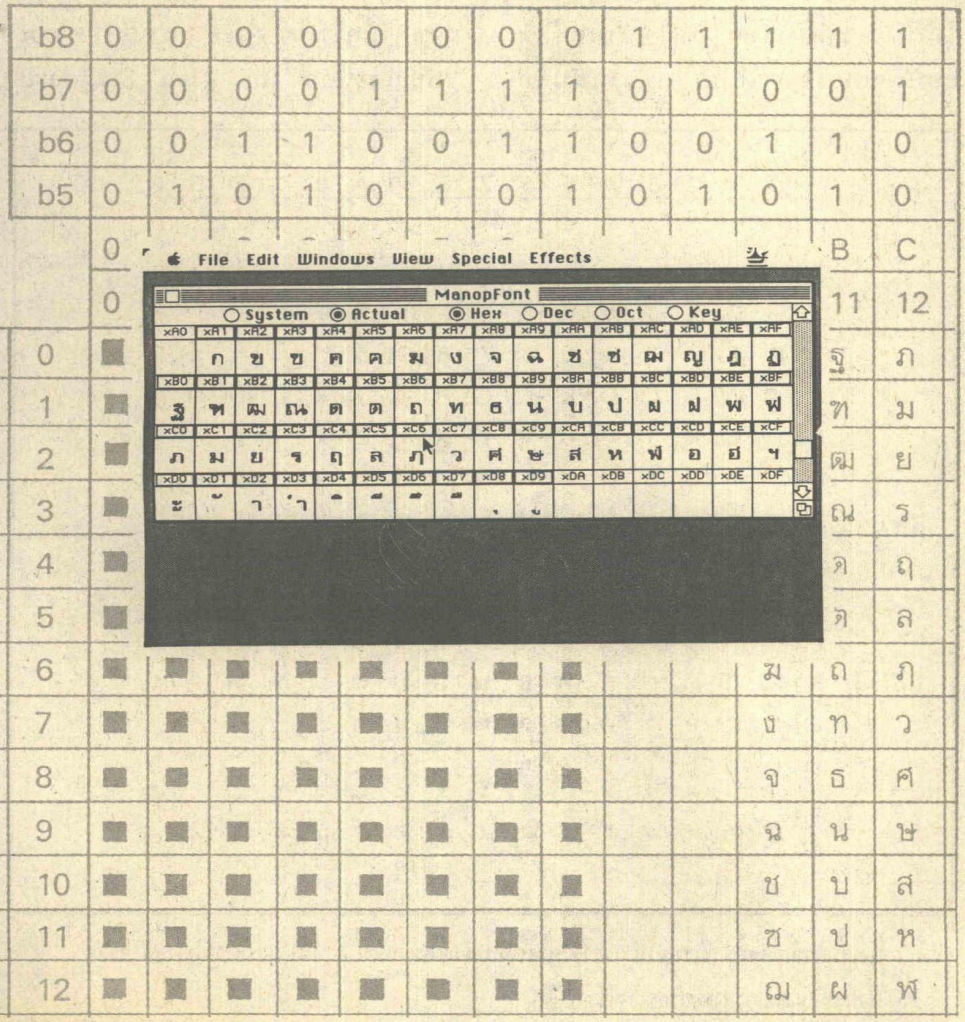
บทความพิเศษ

COMPUTER STANDARDS

ยืน ภู่วรวรรณ

รหัสไทย
กับมาตรฐานระหว่างประเทศ

มาตรฐานของรหัสที่ใช้แทนอักขระไทย เมื่อก้าวเข้าสู่การประยุกต์แบบสากลแล้วจะมีหนทางการพัฒนาต่อไปอย่างไร



พัฒนาการการใช้ภาษาไทยบนเครื่องคอมพิวเตอร์ มีความจำเป็นควบคู่กับการใช้คอมพิวเตอร์ในประเทศไทย การใช้ภาษาไทยได้เปลี่ยนแปลงก้าวหน้าขึ้นเป็นลำดับ มีการสร้างซอฟต์แวร์ที่ใช้ภาษาไทยและประมวลผลภาษาไทยได้เป็นอย่างดี การใช้ภาษาไทยกับคอมพิวเตอร์อยู่บนพื้นฐานของการแทนรหัสอักขระไทย ซึ่งการใช้งานปัจจุบันมิได้จำกัดอยู่ที่ภาษาไทยภาษาเดียว แต่ต้องใช้ร่วมกับอักขระภาษาอังกฤษอยู่ด้วย แต่หากนึกดูว่าถ้าต้องการได้การประมวลผลร่วมมากกว่าสองภาษา เช่น มีเวิร์ดโปรเซสเซอร์ที่พิมพ์ได้ทั้งไทย-อังกฤษ-ฝรั่งเศส หรือญี่ปุ่น จะทำได้อย่างไร ความต้องการของภาษาในโลกนี้มีมากมาย ดังนั้นบทความนี้จึงเน้นให้เห็นแนวทางว่า มาตรฐานของรหัสที่ใช้แทนอักขระไทย เมื่อก้าวเข้าสู่การประยุกต์แบบสากลจริง ๆ แล้ว จะมีหนทางการพัฒนาต่อไปอย่างไร

ความเป็นมาในอดีต

ประเทศไทยเริ่มนำคอมพิวเตอร์มาใช้ครั้งแรกในปี พ.ศ. 2507 คอมพิวเตอร์ยุคแรก ๆ เน้นการประมวลผลข้อมูลที่เป็นรหัสอักขระภาษาอังกฤษ และส่วนใหญ่เป็นเครื่องในตระกูลไอบีเอ็มใช้อักขระรหัสเอปซีดิค (EBCDIC) มีการใช้บัตรเจาะรู

ในยุคของการใช้เครื่องไอบีเอ็มในตระกูล 360/370 ประเทศไทยได้เริ่ม

ใช้ข้อมูลภาษาไทยกันบ้างแล้ว โดยเน้นงานประมวลผลแบบแบตช์ และงานพิมพ์ข้อมูลทางเครื่องพิมพ์ การกำหนดรหัสแทนอักขระจึงเริ่มในลักษณะที่ต่างคนต่างทำตามความเหมาะสมของแต่ละเครื่อง

ในปี พ.ศ. 2520 เริ่มจาก ANSI ได้ตกลงสร้างรหัสแทนอักขระตัวอักษรภาษาอังกฤษจำนวน 128 ตัว และต่อมา ISO ก็รับนำมาเป็นรหัส ISO 646 และแพร่หลายใช้งานกันทั่วโลก

สำหรับในทีมงานของผู้เขียนได้เริ่มสนใจและพัฒนาการใช้ภาษาไทยตั้งแต่ยุคเริ่มต้นของไมโครคอมพิวเตอร์ ในปี พ.ศ. 2521 ทีมงานของผู้เขียนที่ห้องปฏิบัติการวิจัยไมโครคอมพิวเตอร์ได้สั่งไมโครคอมพิวเตอร์เข้ามาใช้เป็นเครื่องแรกสำหรับใช้งาน คือ โมเดล IMSAI 8080 เครื่องในสมัยนั้นต้องต่อกับดัมพ์เทอร์มินัล โปรแกรมที่ใช้คือ CP/M เวอร์ชัน 1.3 และมีภาษา MBASIC ติดมาด้วย

ผู้เขียนให้ความสนใจอยากแก้ไขดัมพ์เทอร์มินัลให้แสดงผลภาษาไทยได้ จึงได้เปลี่ยนรอมเพื่อแสดงผลภาษาไทย การประมวลผลใช้ภาษาเบสิกและแอสเซมบลีเป็นหลัก การดำเนินการทั้งในลักษณะค้นคว้าและทดลอง

ในปี พ.ศ. 2524 ผู้เขียนได้มีโอกาสเดินทางไปทำวิจัยที่สถาบัน National Museum of Ethnology ที่กรุงโอซาก้า ประเทศญี่ปุ่น เป็นเวลาประมาณ 1.5 เดือน การวิจัยครั้งนั้นเน้นการสร้าง Thai text editor และได้นำผลงานเสนอในการประชุมวิชาการวิศวกรรมไฟฟ้าครั้งที่ 4 เดือนพฤศจิกายน 2525

ในปี พ.ศ. 2525 ทีมงานของผู้เขียนได้เสนอผลงานเรื่องการประมวลผลข้อมูลตัวอักษรไทยด้วยไมโครคอมพิวเตอร์ และได้รับรางวัลผลงานคิดค้นและสิ่งประดิษฐ์ที่เป็นประโยชน์ต่อประเทศชาติ รางวัลที่ 2 ในปี พ.ศ. 2526

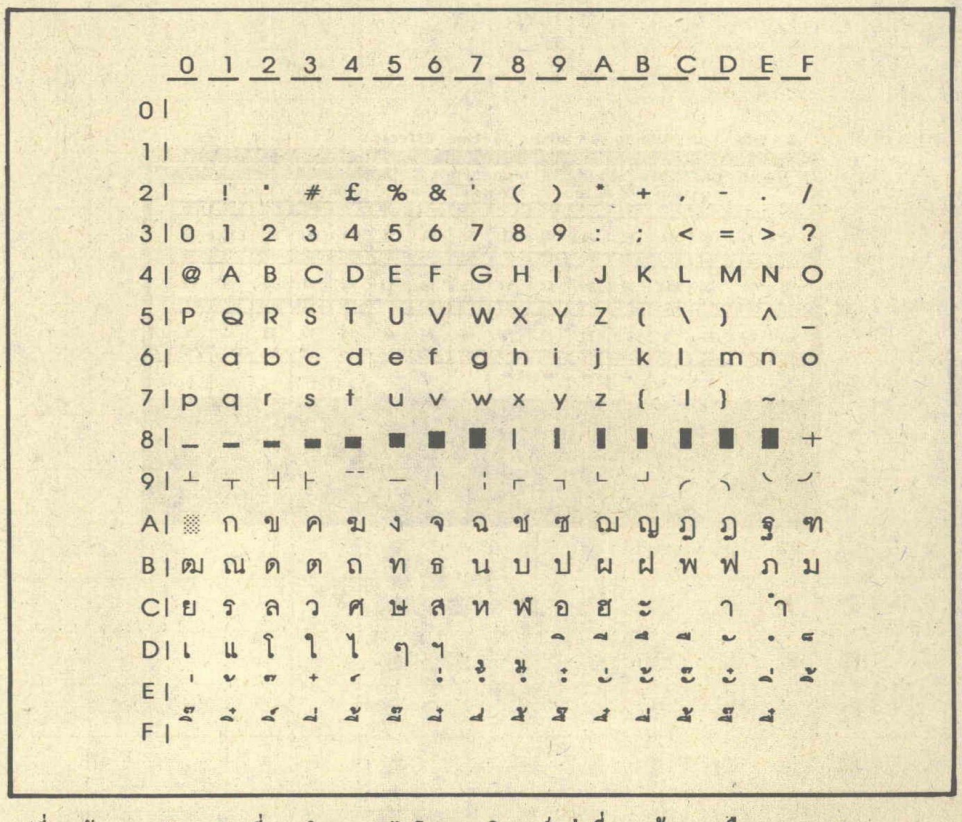
จุดเริ่มต้นของรหัสเกษตร

ในยุคสมัยไมโครคอมพิวเตอร์ที่กำลังเข้ามาในประเทศไทยอย่างมากนั่นเอง ทีมงานวิจัยของห้องปฏิบัติการวิจัยไมโครคอมพิวเตอร์ก็ได้พัฒนาและดัดแปลงเครื่องพิมพ์ดอตแมตริกซ์ และเครื่องคอมพิวเตอร์หลายยี่ห้อให้ใช้กับภาษาไทยได้

ในปี พ.ศ. 2525 คณะผู้ทำวิจัยได้เสนอรหัสภาษาไทยใช้กับเครื่องคอมพิวเตอร์หลายยี่ห้อ หนึ่งในนั้นคือออสบอร์น และได้พัฒนาโปรแกรมเวิร์ดโปรเซสเซอร์ชื่อ "ขวัญใจนักพิมพ์ดีด" ต่อมาได้พัฒนาบนเครื่อง 16 บิต วิกเตอร์และไซเรียส (ดูโฆษณาในวารสารไมโครคอมพิวเตอร์ เล่ม 1, 2, 3,...)

วารสารไมโครคอมพิวเตอร์ ฉบับที่ 4 ปีที่ 1 ตุลาคม 2524 ผู้เขียนได้เสนอบทความเรื่อง "ถึงเวลาที่จะหารหัสมาตรฐานภาษาไทยแล้วหรือยัง" โดยเสนอเหตุผลที่จะต้องพิจารณารหัสมาตรฐาน และชี้ให้เห็นว่าพื้นฐานของรหัสเกษตรพัฒนาอย่างไร

ในการประชุมวิชาการทางวิศวกรรมไฟฟ้า 8 สถาบันอุดมศึกษา ครั้งที่ 5 วันที่ 15 พฤศจิกายน 2525 ที่มหาวิทยาลัยสงขลานครินทร์ ผู้เขียนเสนอบทความ "Suggestion for Thai Standard



รูปที่ 1 ข้อเสนอรหัสเกษตร ที่เสนอในวารสารไมโครคอมพิวเตอร์ เล่มที่ 4 หน้า 98 เดือนตุลาคม 2526 ซึ่งมีรหัสผสมและการตีกรอบเอาไว้

Character Code" บทความกล่าวถึงอัลกอริทึมการแสดงผลภาษาไทยทางจอภาพ ทางเครื่องพิมพ์ การจัดเรียงคำไทย และการประมวลผล ตลอดจนปัญหา โดยเน้นให้เห็นว่า การวางรหัสภาษาไทยจะเกี่ยวข้องกับการประมวลผลอย่างไร

ในปี พ.ศ. 2527 กลุ่มนักวิจัยจากห้องปฏิบัติการวิจัยไมโครคอมพิวเตอร์ มหาวิทยาลัยเกษตรศาสตร์ ได้รับรางวัลผลงานคิดค้นและสิ่งประดิษฐ์ที่เป็นประโยชน์ต่อประเทศชาติ รางวัลที่ 2 เป็นครั้งที่ 2 ในชื่อผลงาน อัลกอริทึมคำไทยและการสังเคราะห์เสียงพูดภาษาไทย ผลงานดังกล่าวมีการวิเคราะห์สถิติที่เกี่ยวข้องกับคำไทย และอัลกอริทึมพื้นฐานที่ใช้ในการประมวลผลข้อมูลภาษาไทย

ในปี พ.ศ. 2529 สำนักงานมาตรฐานผลิตภัณฑ์อุตสาหกรรม ได้เสนอมาตรฐานรหัสสำหรับอักขระไทยที่ใช้กับคอมพิวเตอร์ เคยได้เสนอเป็นมาตรฐานเลขที่ มอก. 622/2529 คณะกรรมการที่ดำเนินการนี้คือ คณะกรรมการที่ 478 ซึ่งก็มีผู้เขียนเป็นผู้ร่วมอยู่ด้วยในฐานะกรรมการร่างคนหนึ่ง

สถานภาพการใช้รหัส สมอ.

ก่อนอื่นขอพิจารณาโครงสร้างอักขระไทยที่ใช้ในการประมวลผล ซึ่งแบ่งแยกการเขียนได้ 4 ระดับ ลักษณะโครงสร้างภาษาไทยเมื่อเขียนตามกฎไวยากรณ์ BNF เขียนได้ดังรูปที่ 2

ตาม มอก.620/2529 ได้เสนอจุดมุ่งหมายไว้ว่าให้เป็นรหัสสำหรับการ interchange จึงไม่รวมรหัสผสมสำหรับเครื่องพิมพ์และการตีกรอบลากเส้นเอาไว้

รายละเอียดของ มอก.620/2529 แยกออกเป็นสองตารางคือ ตารางที่ใช้ร่วมกับแอสกี หรือรหัส ANSI กับตารางที่บริษัทไอบีเอ็มใช้คือเอ็บซีดิก ดังรูปที่ 3

นอกจากนี้ทางสำนักงานมาตรฐานผลิตภัณฑ์อุตสาหกรรม ยังได้กำหนดมาตรฐานการกำหนดตำแหน่งของตัวอักขระไทยบนแป้นพิมพ์คอมพิวเตอร์ โดยกำหนดชื่อมาตรฐานนี้ว่า มอก. 820/2531 ตำแหน่งของแป้นพิมพ์คอมพิวเตอร์ตามมาตรฐานดังกล่าวแสดงดังรูปที่ 4

สถานการณ์หลังจากการประกาศใช้ มอก. 620/2529

การกำหนดมาตรฐานรหัสอักขระภาษาไทยที่ใช้คอมพิวเตอร์ ถือได้ว่าเป็นจุดเริ่มต้นของการสร้างมาตรฐานและแนวทางการใช้งานการประมวลผลภาษาไทยต่อไป

อย่างไรก็ตาม การแทนข้อมูลภาษาไทยจำเป็นต้องแปลงจากตัวอักขระที่มีหลายระดับให้มาอยู่ในรูปแบบสตริง หรือตัวอักขระเรียงกันไปตามรูปแบบการเก็บข้อมูลด้วยคอมพิวเตอร์ ดังเช่น

ผู้ชายคนหนึ่ง = ผ ู ้ชายคนหน ึ่ ง

ข้อมูลตัวอักขระที่เก็บแบบสตริงนี้ไม่มีกฎเกณฑ์การแบ่งแยกคำออกมาได้เหมือนข้อมูลภาษาอังกฤษ ที่ใช้ช่องว่างระหว่างคำ การประมวลผลข้อมูลอักขระไทยนี้จึงต้องการอัลกอริทึม การจัดการพื้นฐานในเรื่องต่าง ๆ ดังนี้

1. การรับข้อมูลจากแป้นพิมพ์แบบภาษาไทย และภาษาอื่นร่วม
2. การแสดงผลตัวอักขระภาษาไทยบนจอภาพ
3. การแสดงผลตัวอักขระผ่านทางเครื่องพิมพ์
4. การจัดเรียงลำดับคำไทย
5. การแบ่งแยกคำไทยเพื่อการ

```
<thai_character>    ::= char_level1|char_level2|char_level3|char_level4
<char_level1>       ::= t_tone_mark
<char_level2>       ::= t_upper_vowel|t_tone_mark
<char_level3>       ::= t_consonant|t-normal_vowel|t_numeric|
                        t_special_mark
<char_level4>       ::= t_lower_vowel
<t_tone_mark>       ::= ่ ้ ๊ ๋
<t_upper_vowel>     ::= ็ ์ ิ ี ึ ื ั ๎
<t_normal_vowel>    ::= ะ า เ แ ไ ใ
<t_consonant>       ::= ก ข ค ฆ ง จ ฉ ช ซ ฌ ญ ฎ ฏ ฐ ฑ
                        ฒ ณ ด ต ถ ท ธ น บ ป ผ ฝ พ ฟ ภ
                        ม ย ร ฤ ล ว ศ ษ ส ห ฬ อ ฮ
<t_numeric>         ::= ๐ ๑ ๒ ๓ ๔ ๕ ๖ ๗ ๘ ๙
<t_special_mark>    ::= ๆ ฯ ๅ ฿ ๏ ๚ ๛
<t_tower_vowel>     ::= ุ ู ฺ
```

รูปที่ 2

ประมวลผล

การประยุกต์ภาษาไทยบนไมโครคอมพิวเตอร์ มีการจัดรูปแบบของการแสดงผลข้อมูลบนจอภาพได้หลายแบบ แต่ละแบบก็มีลักษณะเฉพาะตัว ผู้ประยุกต์หรือสร้างไทยการ์ดจะยึดหลักในแนวทางที่แตกต่างกันออกไป รูปแบบการแสดงผลจึงขึ้นอยู่กับโครงสร้างการออกแบบฮาร์ดแวร์ ส่วนหนึ่งการแสดงผลดังกล่าวแสดงได้หลายรูปแบบ ดังรูปที่ 5

ในการแสดงผลบนจอภาพก่อปัญหาทำให้ซอฟต์แวร์ที่ใช้งานจะเฉพาะเจาะจงลงไปไม่สามารถใช้งานร่วมกันได้ ผู้ใช้ฮาร์ดแวร์แบบใดจะต้องใช้เฉพาะซอฟต์แวร์ที่ได้พัฒนามาในกลุ่มนั้นเท่านั้น

ปัญหาที่เกิดขึ้นที่สำคัญอย่างหนึ่งคือการแสดงผลผ่านทางเครื่องพิมพ์ ซึ่งในปัจจุบันเครื่องพิมพ์ประเภทดอต-แมตริกซ์นิยมที่จะแสดงผลแบบสามระดับผสมสระวรรณยุกต์ ทำให้ตัวอักขระที่จะใช้ต้องเพิ่มเติมขึ้นในรูปแบบการผสมอักขระระหว่างวรรณยุกต์กับสระบน ตัวอักขระทั้งหมดจึงกำหนดได้ดังรูปที่ 6

ปัญหาที่เกิดขึ้นคือ มอก. 620/2529 มิได้กำหนดแนวทางสำหรับการสร้างรหัสผสมดังกล่าว จึงทำให้ผู้พัฒนาแต่ละรายจึงกำหนดกันขึ้นเอง เป็นผลทำให้เครื่องพิมพ์ที่ใช้งานในหน่วยงานเดียวกัน แต่จัดซื้อจากต่างบริษัทไม่สามารถใช้งานได้ ดังที่เป็นปัญหาอยู่ในขณะนี้

					b8	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
					b7	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
					b6	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
					b5	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
						0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
b4	b3	b2	b1			0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	■	■	■	■	■	■	■	■				ฐ	ภ	ะ	เ	๐
0	0	0	1	1	1	■	■	■	■	■	■	■	■			ก	ฑ	ม	ั	แ	๑
0	0	1	0	2	2	■	■	■	■	■	■	■	■			ข	ฒ	ย	า	โ	๒
0	0	1	1	3	3	■	■	■	■	■	■	■	■			ฃ	ณ	ร	ำ	ใ	๓
0	1	0	0	4	4	■	■	■	■	■	■	■	■			ค	ด	ฤ	ิ	ไ	๔
0	1	0	1	5	5	■	■	■	■	■	■	■	■			ฅ	ต	ล	ี	า	๕
0	1	1	0	6	6	■	■	■	■	■	■	■	■			ฆ	ถ	ฦ	ื	ๆ	๖
0	1	1	1	7	7	■	■	■	■	■	■	■	■			ง	ท	ว	ื	็	๗
1	0	0	0	8	8	■	■	■	■	■	■	■	■			จ	ธ	ศ	ุ	่	๘
1	0	0	1	9	9	■	■	■	■	■	■	■	■			ฉ	น	ษ	ู	้	๙
1	0	1	0	A	10	■	■	■	■	■	■	■	■			ช	บ	ส		๊	ๆ
1	0	1	1	B	11	■	■	■	■	■	■	■	■			ซ	ป	ห		+	๚
1	1	0	0	C	12	■	■	■	■	■	■	■	■			ฌ	ผ	ฬ		์	
1	1	0	1	D	13	■	■	■	■	■	■	■	■			ญ	ฝ	อ		๋	
1	1	1	0	E	14	■	■	■	■	■	■	■	■			ฎ	พ	ฮ		๎	
1	1	1	1	F	15	■	■	■	■	■	■	■	■			ฏ	ฟ	ฯ	B	๏	

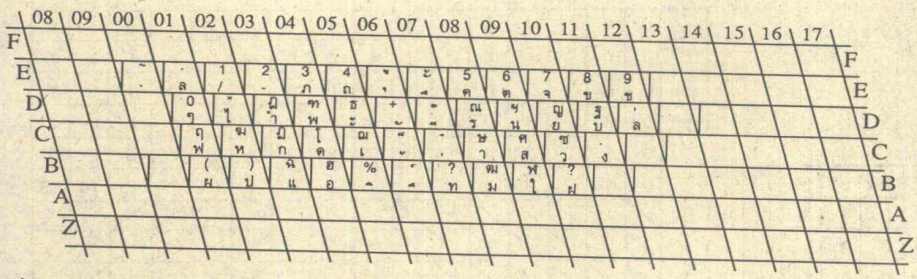
หมายเหตุ ■ หมายถึง ใช้ตาม ISO 646

รูปที่ 3 ก. มอก. 620/2529 ตามตารางแอสกี

					b8	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
					b7	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
					b6	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
					b5	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
						0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
b4	b3	b2	b1			0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0	■	■	■	■	■	■	■	฿	๏	๚	๚	๐	■	■	■	■
0	0	0	1	1	1	■	■	■	■			■	๎	■	■		๑	■	■	■	■
0	0	1	0	2	2	■	■	■	■	ก	จ	ฏ	ถ	■	■	■	๒	■	■	■	■
0	0	1	1	3	3	■	■	■	■	ข	ฉ	ฐ	ท	■	■	■	๓	■	■	■	■
0	1	0	0	4	4	■	■	■	■	ซ	ช	ฑ	ธ	■	■	■	๔	■	■	■	■
0	1	0	1	5	5	■	■	■	■	ค	ซ	ฒ	น	■	■	■	๕	■	■	■	■
0	1	1	0	6	6	■	■	■	■	ต	ฌ	ณ	บ	■	■	■	๖	■	■	■	■
0	1	1	1	7	7	■	■	■	■	ฆ	ญ	ด	ป	■	■	■	๗	■	■	■	■
1	0	0	0	8	8	■	■	■	■	ง	ฎ	ต	ผ	■	■	■	๘	■	■	■	■
1	0	0	1	9	9	■	■	■	■	■	■	■	■	■	■	■	๙	■	■	■	■
1	0	1	0	A	10	■	■	■	■	■	■	■	■	ฝ	ร	ษ	ๆ		■	า	๋
1	0	1	1	B	11	■	■	■	■	■	■	■	■	พ	ฤ	ส	ะ	ี	เ	ๆ	์
1	1	0	0	C	12	■	■	■	■	■	■	■	■	ฟ	ล	ห	ั	ื	แ	็	๐
1	1	0	1	D	13	■	■	■	■	■	■	■	■	ภ	ฦ	ฬ	า	ื	โ	่	
1	1	1	0	E	14	■	■	■	■	■	■	■	■	ม	ว	อ	ำ	ุ	ใ	ั	
1	1	1	1	F	15	■	■	■	■	■	■	■	■	ย	ศ	ฮ	ิ	ู	ไ	๊	

รูปที่ 3 ข. มอก. 620/2529 ตามตารางเอ็บซีดิก

รูปที่ 3 มาตรฐาน มอก. 620/2529



รูปที่ 4 มาตรฐานการกำหนดตำแหน่งแป้นพิมพ์ มอก. 820/2531

ก. แบบ 4 ระดับ

						่								
				้		ี								
ร	ะ	บ	บ	ผู้	เ	ชี	ย	ว	ช	า	ญ			
				ู										

ข. แบบ 3 ระดับผสมสระวรรณยุกต์

				้		ี่					
ร	ะ	บ	บ	ผู้	เ	ชี่	ย	ว	ช	า	ญ
				ู							

ค. แบบ 2 ระดับผสมข้ามบรรทัด

				้		ี่					
ร	ะ	บ	บ	ผู้	เ	ชี่	ย	ว	ช	า	ญ
				ี่			ั				
		เ	ค	รื่	อ	ง	จั	ก	ร		

ง. แบบหนึ่งระดับ (แบบรูปภาพ)

ร	ะ	บ	บ	ผู้	เ	ชี่	ย	ว	ช	า	ญ
		เ	ค	รื่	อ	ง	จั	ก	ร		

จ. แบบแยกตัวพยัญชนะเป็น 2 ส่วน

						่								
				้		ี่								
ร	ะ	บ	บ	ผู้	เ	ชี่	ย	ว	ช	า	ญ			
				ู										

รูปที่ 5 โครงสร้างการจัดแสดงผลภาษาไทยแบบต่าง ๆ

```
<Upper__line__Display>   ::- t__upper__vowels¦t__tone__mark¦combine__char
<Normal__line__Display>  ::- t__normal__vowel¦t__consonant¦
                             t__numeric¦t__special__mark
<lower__line__Display>   ::- t__under__vowel
<combine__char>          ::-
```

รูปที่ 6

จำเป็นต้องเพิ่มรหัส รหัสอักขระไทยผสม และการตีตาราง

เมื่อผู้ใช้ใช้ มอก. 620/2529 มากขึ้น ปัญหาที่เกิดขึ้นตามมา คือผู้พัฒนาโปรแกรมพัฒนาเครื่องพิมพ์ไปคนละแนวทาง โดยไม่มีข้อเสนอแนะใดจากสำนักงานมาตรฐาน ผลที่สุดก็คือ มีรหัส มอก. ที่ใช้ร่วมกันโดยตรงไม่ได้ (ดูจากรหัสเครื่องพิมพ์ที่บริษัทสหวิริยาพัฒนาใส่ในรุ่น EPSON LQ 1050+ หรือดูในบทความเรื่อง EPSON LQ 1050+ หนังสือไมโครคอมพิวเตอร์ ฉบับที่ 58 หน้า 202-207) เกิดปัญหากับผู้ใช้อย่างมาก ในขณะนั้นหากกำหนดมาตรฐานเครื่องคอมพิวเตอร์ และเครื่องพิมพ์เป็นแบบ มอก. 620/2529 แล้วใช่ว่าจะใช้ร่วมกันได้ บนเครื่องพิมพ์ในสำนักงานจึงต้องติดสติ๊กเกอร์บอกว่าเป็น มอก. ตามบริษัทใดอีกขั้นหนึ่ง เพราะรหัสผสม และการตีกรอบ ตีเส้นไม่ตรงกัน

สำนักงานมาตรฐานผลิตภัณฑ์อุตสาหกรรม จึงต้องแก้ปัญหาที่ตามมาอีกขั้นด้วยการปรับปรุง มอก. 620/2529 เป็น มอก. 620/2533 และเพิ่มเติม มอก. 988/2533 เป็นเรื่องข้อแนะนำเกี่ยวกับรหัสสำหรับอักขระไทยผสมและสัญลักษณ์สำหรับการตีตารางสำหรับเครื่องแบบจุด

ลักษณะของ มอก. 988/2533 แสดงได้ดังรูปที่ 7

จะเห็นว่าในที่สุดสำนักงานมาตรฐานผลิตภัณฑ์อุตสาหกรรมต้องเข้ามาแก้ปัญหาเรื่องการตีกรอบลากเส้น และรหัสผสมของเครื่องพิมพ์ เพื่อให้ผู้ปฏิบัติ

ปฏิบัติได้ถูกต้อง และลดปัญหาที่เกิดขึ้นกับผู้ใช้

รหัสภาษาไทยต้องใช้งานร่วมกับทุกภาษาได้ในโลก

นอกจากภาษาอังกฤษที่เป็นภาษาหลักที่คนไทยต้องใช้ร่วมกับภาษาไทยแล้วยังมีภาษาอื่นอีกมากมาย เช่น ครูสอนภาษาฝรั่งเศส อยากเก็บข้อมูลภาษาฝรั่งเศสร่วมกับภาษาไทย ภาษาญี่ปุ่นร่วมกับภาษาไทย ฯลฯ การออกแบบรหัสภาษาไทยบนตาราง ASCII ทำให้เราใช้ร่วมกับภาษาอังกฤษได้ดี แต่ก็เป็นปัญหาสำหรับบางโปรแกรมที่แต่เดิมใช้ระหว่างภาษาอังกฤษกับภาษาทางยุโรปอื่นได้ ด้วยเหตุนี้เองการใช้งานคอมพิวเตอร์ในยุคต่อมาต้องคำนึงถึงการใช้งานกับภาษาทุกภาษาในโลก ซึ่งคาดกันว่ามีถึง 4,000 ภาษา

บางภาษาเช่นกลุ่มภาษาทางตะวันออก จีน ญี่ปุ่น มีการจัดวางรหัสออกเป็นสองไบต์ เพื่อจะใช้แทนตัวลักษณะรูปภาพ ดังตัวอย่างรหัสภาษาญี่ปุ่นที่ยกตัวอย่างให้ดูเป็นบางส่วน ดังรูปที่ 8

หากต้องการใช้ภาษาไทยร่วมกับซอฟต์แวร์ของภาษาญี่ปุ่น ดังเช่นโครงการภาษาที่กระทรวงวิทยาศาสตร์ โดยศูนย์อิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ ซึ่งในโครงการจำเป็นจะต้องพัฒนาการป้อนข้อมูลภาษาไทยเข้าในซอฟต์แวร์ของประเทศญี่ปุ่น ดังนั้นหากโครงสร้างภาษาไทยแทนแบบรูปภาพเหมือนภาษาญี่ปุ่นก็จะจัดการด้านการแสดงผลได้โดยไม่ต้องแก้ไขอะไรมาก

โครงสร้างของรหัสที่ใช้จึงควรมีการแสดงผล แบบหนึ่งระดับ (แบบรูปภาพ) เช่น กู้ เป็นรูปภาพหนึ่ง กล่าวคือ

	b8	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
	b7	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
	b6	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
	b5	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
b4	b3	b2	b1			0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	0	0									฀	๐						
0	0	0	1	1	1									ก	๑						
0	0	1	0	2	2									ข	๒						
0	0	1	1	3	3									ฃ	๓						
0	1	0	0	4	4									ค	๔						
0	1	0	1	5	5									ฅ	๕						
0	1	1	0	6	6									ฆ	๖						
0	1	1	1	7	7									ง	๗						
1	0	0	0	8	8									จ	๘						
1	0	0	1	9	9									ฉ	๙						
1	0	1	0	A	10									ช	ฯ						
1	0	1	1	B	11									ซ	๚				฿		
1	1	0	0	C	12									ฌ	๛				เ		
1	1	0	1	D	13									ญ	็				แ		
1	1	1	0	E	14									ฎ	๎				โ		
1	1	1	1	F	15									ฏ	๏				ใ		

หมายเหตุ 1. ▨ หมายถึง ใช้ตาม ISO 646

2. ▧ หมายถึง ใช้ตามมาตรฐานผลิตภัณฑ์อุตสาหกรรม รหัสสำหรับอักขระไทยที่ใช้กับคอมพิวเตอร์ มาตรฐานเลขที่ มอก.620

รูปที่ 7 รหัส มอก. 988/2533

		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
力	3230													下	化	仮	何
	3240	伽	価	佳	加	可	嘉	夏	嫁	家	寡	科	暇	果	架	歌	河
	3250	火	珂	禍	禾	稼	箇	花	苛	茄	荷	華	菓	蝦	課	嘩	貨
	3260	迦	過	霞	蚊	俄	峨	我	牙	画	臥	芽	蛾	賀	雅	餓	駕
	3270	介	会	解	回	塊	壊	廻	快	怪	悔	恢	懐	戒	拐	改	
	3320		魁	晦	械	海	灰	界	皆	絵	芥	蟹	開	楷	貝	凱	劾
	3330	外	咳	害	崖	慨	概	涯	碍	蓋	街	該	鎧	骸	浬	馨	蛙
	3340	垣	柿	蠣	鈎	割	嚇	各	廓	拡	攪	格	核	殻	獲	確	穫
	3350	覚	角	赫	較	郭	閣	隔	革	学	岳	楽	額	顎	掛	笠	樫
	3360	橿	梶	鰍	潟	割	喝	恰	括	活	渇	滑	葛	褐	轄	且	鰹
	3370	叶	椛	樺	鞄	株	兜	竈	蒲	釜	鎌	噛	鴨	栢	茅	萱	
	3420		粥	刈	苅	瓦	乾	侃	冠	寒	刊	勘	勧	巻	喚	堪	姦
	3430	完	官	寛	干	幹	患	感	慣	憾	換	敢	柑	桓	棺	款	歓
	3440	汗	漢	澗	灌	環	甘	監	看	竿	管	簡	緩	缶	翰	肝	艦
	3450	莞	観	諌	貫	還	鑑	間	閑	関	陥	韓	館	舘	丸	含	岸
	3460	巌	玩	癌	眼	岩	翫	贋	雁	頑	顔	願					

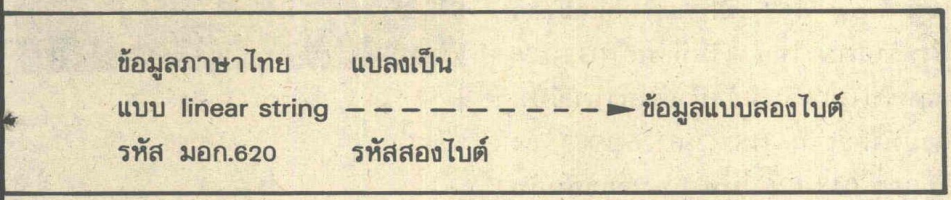
รูปที่ 8 แสดงรหัส 2 ไบต์บางส่วนของภาษาญี่ปุ่น

มีการรวมกันระหว่างสระบน วรรณยุกต์ และสระล่าง ดังรูปที่ 9

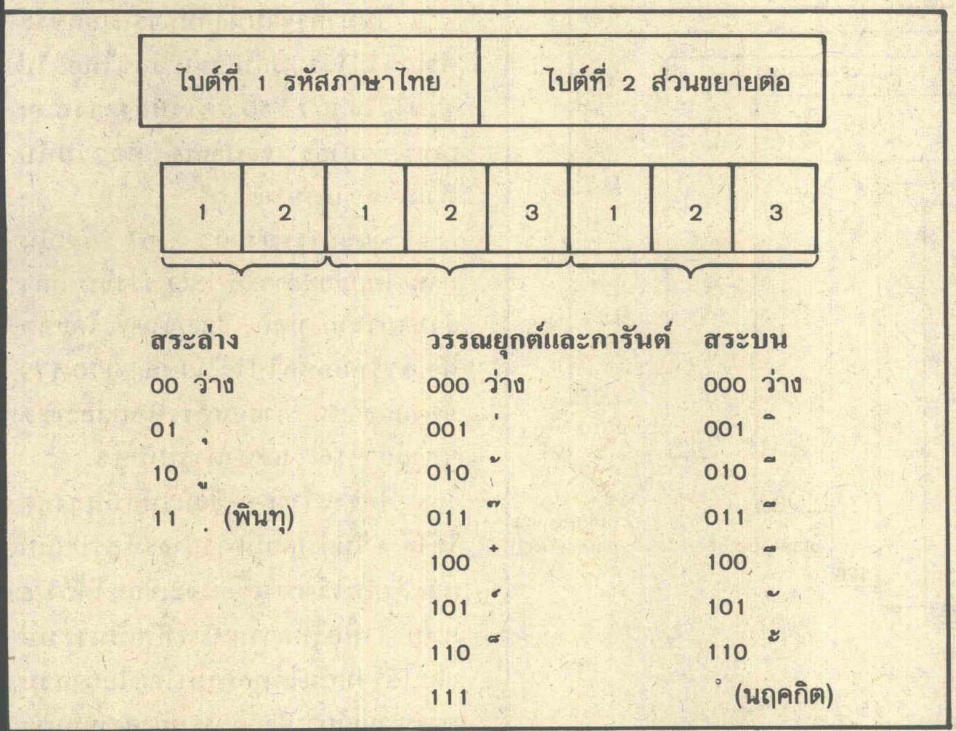
การขยายต่อรหัส มอก. 620 นี้ สามารถทำได้ โดยให้ไบต์เลขคู่เป็นรหัส มอก. เฉพาะส่วนที่เป็นอักขระระดับปกติ ส่วนไบต์เลขคี่จะแทนกลุ่มของรหัสที่กำหนดขึ้นใหม่มาจากสระบน วรรณยุกต์ และสระล่าง ดังโครงร่างที่แสดงดังรูปที่ 10

การสร้างรหัสแบบสองไบต์กรณีนี้เน้นเฉพาะสำหรับใช้กับซอฟต์แวร์ที่มาจากภาษาที่มีโครงสร้างไบต์ เช่น ภาษาญี่ปุ่น เท่านั้น นอกจากนี้ยังนำมาใช้สำหรับเครื่องพิมพ์ที่จะพัฒนาขึ้นในอนาคตให้พิมพ์ภาษาไทยครั้งละหนึ่งบรรทัด โดยไม่ต้องพิมพ์สระบนรอบหนึ่ง พยัญชนะปกติรอบหนึ่ง และสระล่างอีกรอบหนึ่ง อย่างไรก็ตาม การใช้รหัสเดิมของ มอก. ก็สามารถแปลงไปมากับรูปแบบแบบสองไบต์นี้ได้ง่าย

นอกจากนี้กลุ่มผู้เขียนได้ทำการวิจัยในเรื่องเครื่องพิมพ์ เพื่อที่จะสร้างรอมตัวอักษร เพื่อให้สามารถพิมพ์ตัวอักษรในลักษณะแถวเดียว ครบทั้ง 4 ระดับ โดยแทนรหัสภายในลักษณะสองไบต์ เพื่อลดการพิมพ์เหมือนภาษาญี่ปุ่น ผลปรากฏว่าจะได้ตัวอักษรที่ใช้งานจริง ประมาณ 2,800 ตัวอักษร ซึ่งสามารถแทนรอมลงในเครื่องพิมพ์เพื่อให้พิมพ์งานที่เร็วขึ้นกว่าเดิมได้ (ปัจจุบันมีเครื่องพิมพ์ที่พิมพ์ตัวอักษรญี่ปุ่นที่มีหัวพิมพ์ถึง 36 จุด ซึ่งพอเพียงพิมพ์ภาษาไทยได้ทันที)



รูปที่ 9



รูปที่ 10 แสดงการขยายต่อเพื่อเพิ่มอีก 1 ไบต์ในกรณีใช้รหัส 2 ไบต์

มาตรฐาน ISO/DP 10646 กำลังดำเนินการกับรหัสนานาชาติเพื่อให้ใช้ร่วมกันได้

เมื่อต้นปี ค.ศ. 1989 องค์การว่าด้วยเรื่องมาตรฐานระหว่างประเทศ หรือเรียกย่อ ๆ ว่า ISO ได้เสนอฉบับร่างของมาตรฐาน ที่ชื่อ multiple-code character set โดยมีขอบเขตของการสร้างมาตรฐานกลาง สำหรับการใช้รหัสกับทุกภาษาที่ใช้ในโลกนี้ให้ใช้งานร่วมกันได้ ซึ่งภาษาไทยก็เป็นส่วนหนึ่งของร่างฉบับนี้ด้วย

การกำหนดรหัสตามร่างมาตรฐานนี้ เป็นการวางโครงร่างรหัสเป็นแบบ octet คือขนาด 8 บิต และ multiple-octet code character set หมายถึงรหัสตัวอักขระที่ใช้เลขไบนารีแทนอย่างน้อยสองไบต์หรือมากกว่า การจัดวางรหัสจะควบคุมถึงรหัสทุกรูปแบบทั้งที่เป็นอักขระพิเศษเฉพาะอักขระภาษาต่าง ๆ ที่ใช้ในโลก อักขระรูปแบบกราฟิกต่าง ๆ และเครื่องหมายพิเศษก็รวมอยู่ในนี้ด้วยเช่นกัน

โครงสร้างของรหัสที่ได้รับการกำหนดแบ่งออกเป็น group, plane, row และ cell โดยการแทนแต่ละส่วนด้วยรหัส 1 ไบต์ ดังนั้นจึงมีได้ 256 กรุ๊ป แต่ละกรุ๊ปจะมี 256 รหัส (cell) รหัสที่จะแทนได้จากทั่วโลกจึงเป็น

256 × 256 × 256 × 256 รหัส

รหัสแต่ละกรุ๊ปจึงมองเห็นเป็นตารางแบบสามมิติรูปที่ 11

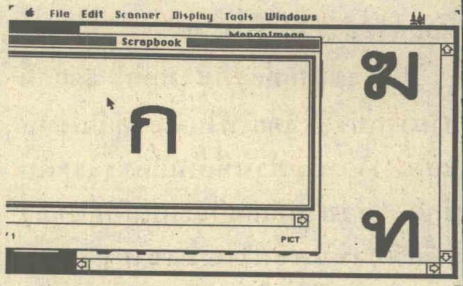
แต่ละกรุ๊ปจะมีเพลน 0322 เป็นเพลนที่ชื่อว่า basic multi-lingual plane เช่นจากตัวอย่างรูปที่ 10 ยังแบ่งออกเป็น 4 กลุ่ม A00, A01, A10, A11, A00 จะตามมาตรฐาน IISXO208: 1983 ซึ่งมีขนาด 79 × 95 เซล A10 จะตามมาตรฐาน GB23122 : 1980 ของ 1980 ของประเทศจีน ซึ่งมีขนาด 80 × 95 เซล ส่วน A11 เป็นรหัสภาษาเกาหลี ตามมาตรฐาน SC5601 : 1987 มีขนาด 80 × 96 เซล

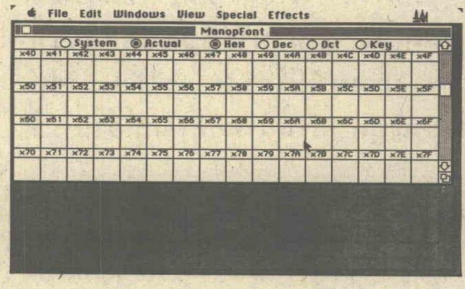
กลุ่มดังกล่าวนี้เป็นเพลนของการใช้รหัสสองไบต์ ซึ่งแต่ละเพลนจะแบ่งแยกออกเป็นโซนดังกล่าวแล้ว

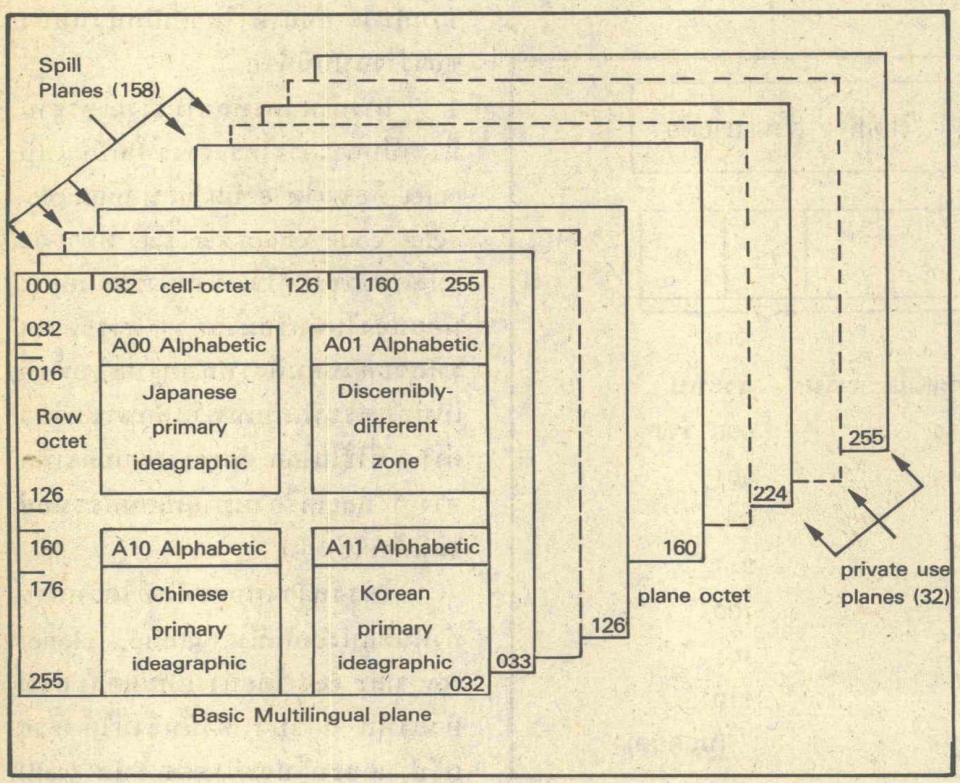
สำหรับกรุ๊ปบางกรุ๊ปมีการจัดวางไว้สำหรับรหัสพื้นฐานแบบหนึ่งไบต์ โดยใน basic plane จะรวมกลุ่มของตัวอักขระของหลาย ๆ ประเทศไว้รวมกันได้ ซึ่งถ้าพิจารณาดูจะเห็นว่า รหัสในหนึ่งเพลนมีได้ถึง 65536 รหัส ซึ่งถ้านำรหัสแบบหนึ่งไบต์มาวางลงจะได้มากมายหลายร้อยภาษาก็ได้

เพลนพื้นฐานของกรุ๊ปนี้คือเพลน 032 รวมภาษาต่าง ๆ ไว้ ซึ่งในกรณีของญี่ปุ่นจะมีฮิรากานะ และ คาตะกานะ อยู่ในเพลนนี้ด้วย สำหรับภาษาไทยและเพื่อนบ้านก็อยู่ในเพลนนี้เช่นกัน ตัวอย่างการจัด basic multi-lingual plane 032 แสดงดังรูปที่ 12

สังเกตว่าในแต่ละแถวมีสองภาษา เช่น row 167 มีภาษาไทยและลาว สำหรับภาษาไทย เริ่มมีเซล 032-126 สำหรับในเพลนนี้ยังนำเอาภาษาญี่ปุ่นตามมาตรฐาน IIS C 6226-1983 ลงในแถว 048-126 ในทำนองเดียวกันกับภาษาจีนเกาหลี





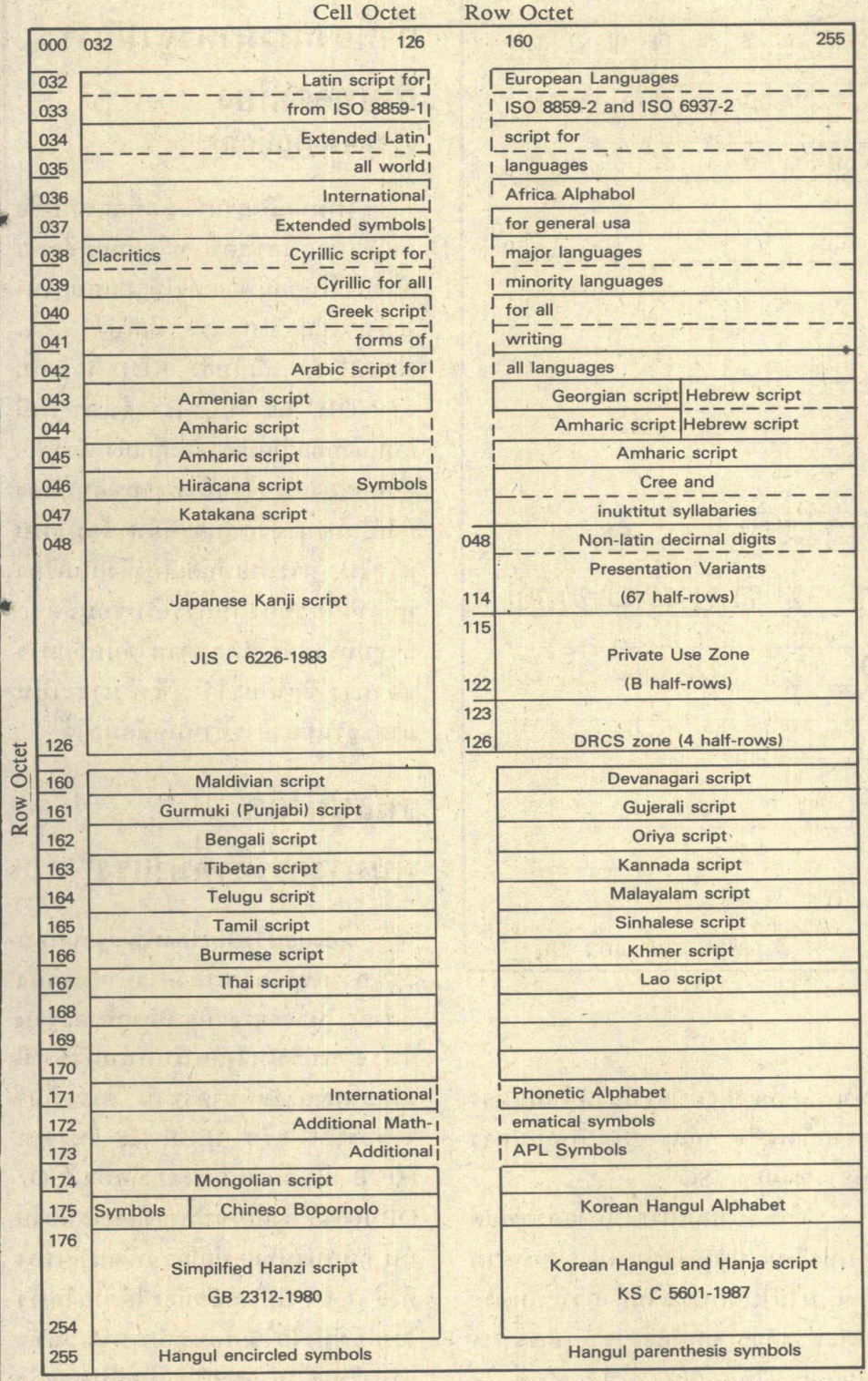


รูปที่ 11 ตัวอย่างการแทนรหัสแบบ multiple octet coded character set

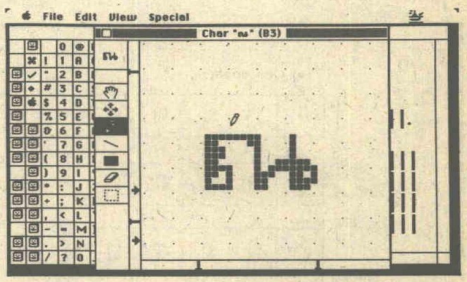
ในมาตรฐานนี้ได้มีการนำอักขระตัวเลขไปไว้รวมกัน เช่น เลขไทย ไปรวมไว้ในแถว 50 ครึ่งหลังตรงส่วน presentation variants คือรวมกับตัวเลขของภาษาอื่น ๆ

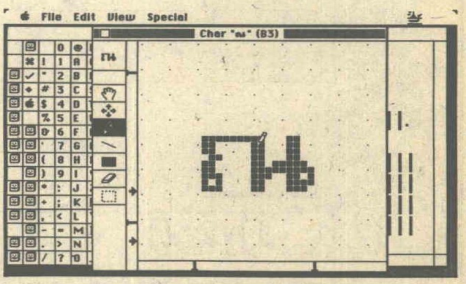
ตัวอย่างของแถว 167 ซึ่งเป็นภาษาไทยและลาว ที่ ISO ร่างขึ้น ก็คงยังนำมาจาก มอก. 620/1986 โดยตัดตัวเลขไทยออกไปไว้ในเซล 170-179 ของแถว 50 การแสดงรหัสแต่ละเซลของแถว 167 แสดงดังรูปที่ 13

โครงสร้างการเขียนรหัสนั้นสูงสุดใช้ได้ 4 ไบต์ โดยมีการวางรหัสรูปแบบและลำดับไว้ การเขียนจะเขียนได้ถึง 5 แบบ เพื่อให้การแทนรหัสเป็นระบบและใช้ร่วมกันได้ทุกภาษาโดยไม่กำกวม มาตรฐานเน้นว่าตัวอักขระหนึ่งความหมายจะต้องอยู่ได้เพียงเซลเดียวเท่านั้น



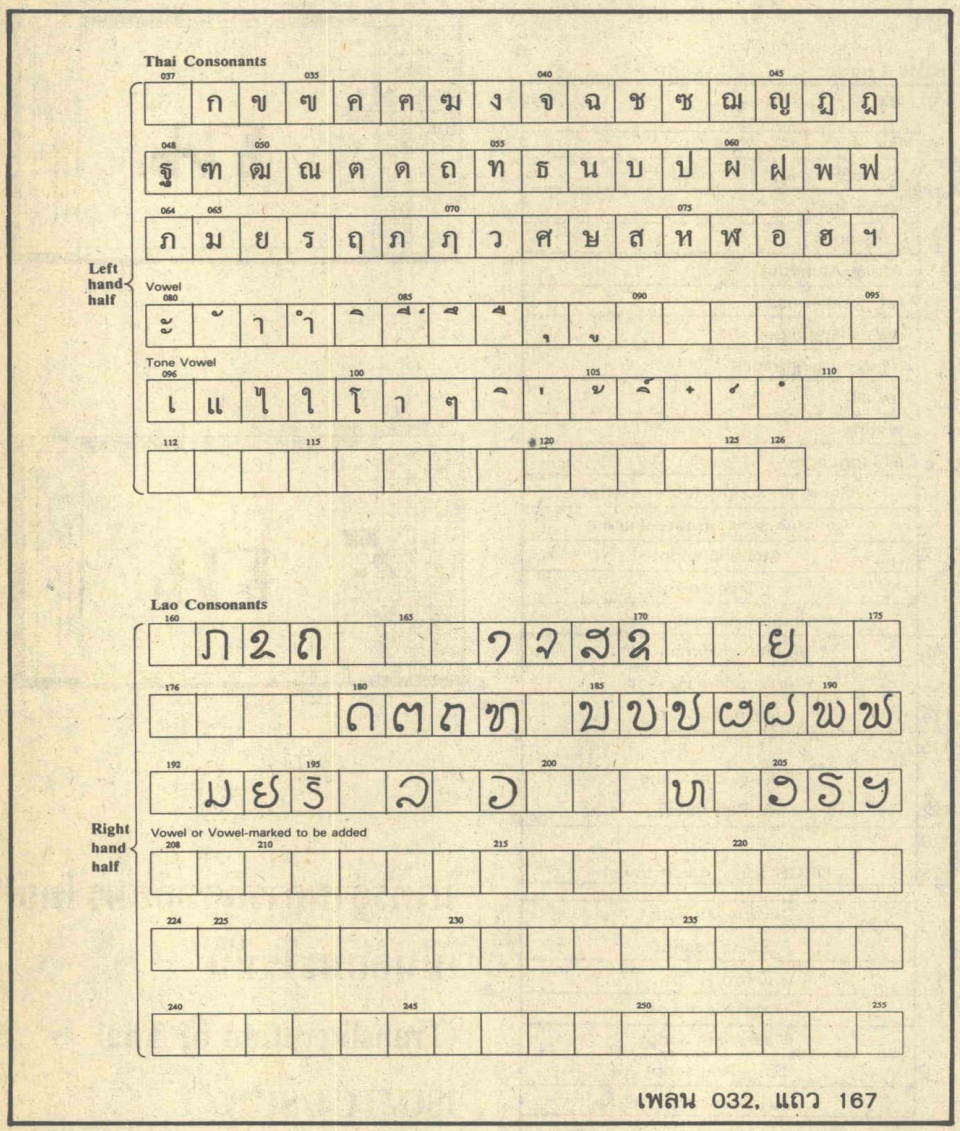
รูปที่ 12 basic multi-lingual plane 032





มาตรฐานการถอดอักษรไทยเป็นอักษรโรมัน (Transliteration of Thai ISO/TC4/SC2)

ปัจจุบันการแลกเปลี่ยนข้อมูลข่าวสารถึงกันเป็นไปอย่างรวดเร็วและเพิ่มมากขึ้น ในห้องสมุดต่างประเทศ เช่น ในประเทศอังกฤษ สหรัฐอเมริกา ฝรั่งเศส ฯลฯ มีการเก็บหนังสือและเอกสารสำคัญที่เป็นภาษาไทยไว้เป็นจำนวนมาก การใช้งานต่างภาษาของแต่ละประเทศก็มีความจำเป็นมากยิ่งขึ้น การเก็บรวบรวมข้อมูลด้วยคอมพิวเตอร์มีความจำเป็น เช่น ห้องสมุดรัฐสภาอเมริกาเก็บหนังสือและเอกสารจากเกือบทุกชาติในโลก ดังนั้นจึงจำเป็น



รูปที่ 13 รหัสสำหรับภาษาไทย และลาว

ต้องถ่ายทอดอักขระภาษาอื่นให้เป็นอักขระโรมัน ซึ่งคอมพิวเตอร์ในปัจจุบันจะประมวลผลภาษาโรมันได้ดีที่สุด ดังนั้นองค์การว่าด้วยเรื่องมาตรฐานระหว่างประเทศ จึงเข้ามาเกี่ยวข้องและพยายามจะทำมาตรฐานการ Transliteration ของทุกภาษาให้เป็นภาษาโรมัน สำหรับภาษาไทย ทางสำนักงานมาตรฐานผลิตภัณฑ์อุตสาหกรรมได้มอบหมายให้ราชบัณฑิตยสถานเป็นผู้จัดทำเสนอต่อ ISO ปัจจุบันได้ดำเนินการร่างแล้วเสร็จ และอยู่ในขั้นตอนการพิจารณาของ ISO

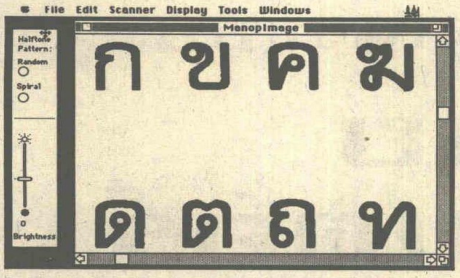
สำหรับหลักการในการถ่ายทอดอักษรไทยไปเป็นโรมันนั้น พยายามให้เป็นไปในลักษณะไม่กำกวม เหมาะเจาะตรงตัว ถูกต้องตามลักษณะการเขียนแบบไทย เช่น ก ใช้ K ข ใช้ Kh เป็นต้น

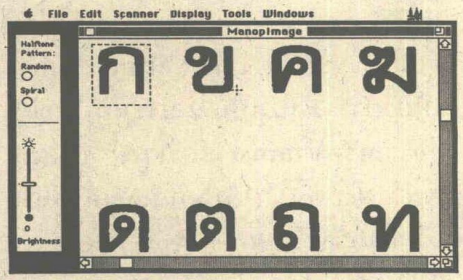
การใช้งานประมวลผลข้อมูลยังต้องการมาตรฐานการถ่ายทอดเสียง (Transcription)

หากนึกถึงฐานข้อมูลที่ต้องเก็บชื่อถนน ชื่อภูมิศาสตร์ หรือแม้แต่ชื่อคนเข้าไปไว้ในคอมพิวเตอร์ในรูปแบบภาษาโรมัน เช่น ปานะเระ ก็เก็บว่า PANARE กุย ก็เก็บว่า KUI งิ้วก็เก็บว่า NGIU คราวนี้ถึงคราวค้นหา ถ้ามีคนหนึ่งบอกว่าตนเองชื่อ ก๊กเม้ง ต้องการข้อมูลของตน เจ้าหน้าที่ฐานข้อมูลต้องคีย์เป็นภาษาอังกฤษจะคีย์หาข้อมูลพบหรือไม่ การถ่ายทอดเสียงจึงเป็นเรื่องหนึ่งที่เกี่ยวข้องกับการเก็บข้อมูลด้วย ปัจจุบันราชบัณฑิตยสถานได้มีหลักการของการเขียนเอาไว้ ซึ่งก็น่าจะเป็นมาตรฐานของการกำหนดข้อมูลได้

รหัสยูนิโค้ดหนทางการค้าที่มาแรง

ต้องยอมรับในเรื่องมาตรฐานก่อนว่า การแข่งขันในเรื่องการค้าเป็นเรื่องที่สำคัญในวงการธุรกิจ หลายมาตรฐานก็มาจากบริษัทโดยเน้นกลุ่มผู้ใช้ที่มีจำนวนมาก เช่นมาตรฐาน 802 ก็มาจากบริษัท หรือ IEEE 488 ก็มาจาก HPIB เป็นต้น เมื่อ ISO คิดทำ ISO/DP 10646 ซึ่งเป็นฉบับร่างและเสนอไปนั้น กลุ่มบริษัทแอปเปิ้ลซึ่งพัฒนาเครื่องที่ใช้ GUI เป็นหลักและได้ขายไปทั่วโลก มีผู้นำไปใช้เกือบทุกภาษา รวมทั้งภาษาไทยด้วย แอปเปิ้ลแมคอินทอชจึงเน้นในเรื่องงานพิมพ์ที่ต้องเกี่ยวพันกับหลายชาติหลายภาษา จึงรวมกลุ่มกันที่





จะพัฒนารหัส multilingua ของตนเอง และตั้งชื่อว่า ยูนิโค้ด

ยูนิโค้ดใช้รหัสเพียงสองไบต์ หรือจำนวน 65536 เซล เพื่อแทนตัวอักษรที่ใช้ในโลกนี้ โดยกลุ่มผู้พัฒนาเห็นว่าน่าจะพอเพียงโดยแบ่งความคิดที่จะรวมรหัสที่มีรูปเหมือนกันไว้ที่เซลเดียว ข้อเด่นจึงทำให้รหัสที่ใช้คงที่สองไบต์ตลอด

วิจารณ์ ISO/DP 10646 กับยูนิโค้ด

จากหลักการของ ISO/DP 10646 ที่องค์การว่าด้วยมาตรฐานระหว่างประเทศร่างขึ้นนี้ เน้นให้รหัสมาตรฐานของแต่ละภาษาเป็นเซตย่อย เช่น ของญี่ปุ่นใช้ JIS C 6266-1983 เป็นเซตย่อย ของจีนใช้ GB 2312-1980 ของไทยใช้ มอก. 620/1986 หากจะพิจารณาที่เพลน 032 ซึ่งเป็นรหัสสองไบต์ จะพบว่ารหัสภาษาที่สำคัญจะมีอยู่ในเพลนนี้เกือบหมดสิ้น เพลน 032 จึงเป็นเพลนหลัก เพลนนี้จึงใช้ออปเซตในไบต์ที่สอง เช่น ของไทยใช้ออปเซต 167 หรือไบต์แรกเป็นรหัส 167

ส่วนรหัสยูนิโค้ดจึงเสมือนย่อส่วนของสี่ไบต์เหลือเพลน 032 เพลนเดียว แล้วพยายามรวมทุกภาษาไว้ในเพลนนี้ให้หมด หากพิจารณาให้ดีแล้ว ภาษาที่ใช้และมีความสำคัญต่อการค้าในช่วงไม่กี่ปีนี้ คงไม่เกินกว่าเพลน 032 ที่ ISO เสนอแน่ ดังนั้น 65536 เซลจึงน่าพอเพียงสำหรับยูนิโค้ด ยิ่งยูนิโค้ดพยายามออกแบบให้ลดตัวซ้ำซ้อนลง 65536 ก็คงจะพอใช้ได้ไปอีกนาน

ISO พยายามมองแบบเผื่อไว้ข้างหน้า เพราะจากการประเมินคาดว่าทั่วโลกมีภาษาประมาณ 4,000 ภาษา บางภาษาเป็นภาษาของชนกลุ่มน้อย บางภาษาก็ไม่มีตัวอักษรของตนเอง ดังนั้นบทบาทของภาษาชนกลุ่มน้อยจึงยังไม่เด่นชัดนัก

ในส่วนของ ISO 10646 นั้น ยังไม่รวมวิธีการใช้รหัสคำสั่งเปลี่ยนเพลน หรือวิธีการเปลี่ยนจาก 1 ไบต์ไปสองไบต์ หรือไป 4 ไบต์ การสวิตช์ระหว่างเพลนเป็นเรื่องที่จะต้องกระทำ เพราะจะใช้รหัสตัวอักษรแต่ละตัวที่ความยาวคงที่ 4 ไบต์คงไม่ดีแน่ แต่ถ้าใช้ได้ตั้งแต่ 1-4 ไบต์ โดยใช้วิธีการสวิตช์ก็จะทำให้การใช้งานสะดวกรวดเร็วในการรับส่งการแสดงผล

อย่างไรก็ตาม ตามฉบับร่างเดิมของไทย มอก. 620/1986 อยู่ที่แถว 167 ตอนล่าง ได้เปลี่ยนเป็นแถว 167 ตอนบนสลับกับลาวแล้ว ทั้งนี้เพราะมีการเสนอแนะจากคนไทยที่มองเห็นว่ารหัสภาษาไทยน่าอยู่ในกลุ่ม 8 บิต 128 รหัสหลังเหมือนตารางแอสกีมากกว่า

อนาคต... มาตรฐานที่เกี่ยวกับประเทศไทย

ตราบใดก็ตามที่เรายังใช้ภาษาไทย จะยังมีมาตรฐานอีกมากที่เกี่ยวข้องกับภาษาไทยที่พวกเราคนไทยจะต้องคิดและยื่นตัวเองเข้าไปมีบทบาทในเวทีมาตรฐานโลก เช่น มาตรฐาน ISO/TC4/SC2 เป็นเรื่องของ Transliteration of Thai มาตรฐานเกี่ยวกับฟอนต์ หรือตัวอักษรมาตรฐานเกี่ยวกับบาร์โค้ดที่เกี่ยวข้องกับภาษาไทย มาตรฐานเกี่ยวกับการเชื่อมโยงโปรแกรม หรือที่เรียกว่า API มาตรฐานเกี่ยวกับการแบ่งแยกคำ มาตรฐานเกี่ยวกับรหัสและการประมวลผลสารสนเทศภาษาไทย

มาตรฐานจึงเป็นเรื่องที่สำคัญมาก เรามักจะรอให้เกิดปัญหาแล้วค่อยแก้ไข หรือว่าเรายังไม่เห็นความสำคัญของมาตรฐาน